

## **A DIAGNOSTIC REPAIR SYSTEM AND METHOD FOR COMPUTING SYSTEMS**

### **BACKGROUND OF THE INVENTION**

#### **Field of the Invention**

[0001] The present invention relates generally to computer diagnostic systems and particularly, to a computer diagnostic system that executes under one operating environment (e.g., Linux) to gather information, diagnose, and make corrections to a second operating environment (e.g., Windows) on the same machine.

#### **Description of the Prior Art**

[0002] Often in the case of computer system failure, a diagnostic software package is executed to attempt to diagnose the cause of the failure. This software may be hosted in its own operating environment (e.g., Linux) outside of the operating system (O/S) that is normally executed on the machine (e.g., Windows). However, unless the diagnostic operating environment is the same as the one being diagnosed and repaired it will not be able to examine the internal configuration of the failed system image. This is because the configuration of the failed system (e.g., Windows) must be manipulated via proprietary interfaces exposed by the failed operating system which are not available in the diagnostic environment (e.g., Linux). For example, consider the case when the diagnostic software is running under Linux while the system to be diagnosed and repaired is a Windows system. This limits the types of repairs the diagnostic software may make. In many cases running the diagnostics software under the same operating system may not be practical for a number of reasons including: image size limitations, cost of additional licenses, or business decisions that do not allow the two O/Ss to be identical.

[0003] One proposed solution for a computer diagnostic system that executes under one operating environment (e.g., Linux) to gather information, diagnose and make corrections to a

second operating environment (e.g., Windows) on the same machine is to write a plethora of Windows diagnostic tools to run under Linux. For example, Linux based tools to manipulate the Windows registry (e.g., Regedt32), event log, INF files, user profiles, Active Directory database files, Security Descriptors, etc. This is a large complex programming and reverse engineering task as the physical layout of these entities on disk are Microsoft proprietary and often change with each update of Windows operating system (O/S). Moreover, there are always new facilities in upcoming releases of Windows O/S (e.g., .NET related system components) that vendors may not be privy to. This will require increased programming and reverse engineering effort to enable diagnosis via Linux based diagnostic tools.

[0004] It would be highly desirable to implement a system that facilitates diagnosing a failed computer system that obviates the complex programming and engineering tasks.

[0005] It would be highly desirable to provide a service that facilitates diagnosing a failed computer system without complex programming and engineering tasks.

### **SUMMARY OF THE INVENTION**

[0006] The present invention relates to a method, computer program product and system for diagnosing and repairing computer systems. The invention is preferably aimed at circumventing the complex programming and engineering task issues associated with current diagnostic and repair methods, by using one or more proxy machines that run the same operating environment as the failed one. A diagnostic software application (e.g., Linux) running on a first computing platform sends relevant information to these proxy machines via a network connection (for example, if available) or through a slower mechanism for diagnostic and repair instructions.

[0007] Thus, according to the invention, there is provided a system and method for diagnosing and repairing computer systems. The system comprises a first computing system executing an operating system of a first platform type and subject to a failure diagnosis; an operating system of a second platform type adapted to be executed on the first computing system

upon failure of the operating system of the first platform type; an application running under the operating system of the second platform type for obtaining diagnostic information pertaining to the failed operating system on the computing system; a second computing system executing the same operating system of the first platform type as the failed operating system executed on the first computing system; a means for communicating the diagnostic information obtained by the application to the second computing system; a means at the second computing system for utilizing the diagnostic information to diagnose the subject failed operating system of the first computing system; and a means executing at the second computing system for generating repair information for communication to the application running on the first computer system to repair the failed operating system.

[0008] Advantageously, the diagnostic repair system is configurable as a service provided by an entity to diagnose and repair remotely located computing systems having failed operating systems.

### **BRIEF DESCRIPTION OF THE DRAWINGS**

[0009] The objects, features and advantages of the present invention will become apparent to one skilled in the art, in view of the following detailed description taken in combination with the attached drawing Figure 1 that illustrates generally the diagnostic repair system 10 according to the invention.

### **DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS**

[0010] Upon a computer system crash, e.g., an operating system (O/S) failure due to data corruption, a virus, or any other conceivable recoverable error (hardware or software), for example, the present invention provides a relatively quick and easy diagnostic solution.

[0011] According to the present invention, a diagnostic software application running on the same computing system as the failed O/S, is provided with read/write access to a file system of

the failed O/S image. Using this capability, the diagnostic software application collects relevant configuration information and error logs from the file system of the failed O/S image, and transports this information to a proxy system running the same operating system as the computing machine being diagnosed. The proxy system utilizes the collected data to diagnose the subject failed O/S system. Once the proxy makes a determination it will synthesize a set of instructions to be transported back to the diagnostic software system to apply. These instructions may include, but are not limited to: copying newly provided files or, overwriting existing data with modified files. It is understood that other possible instructions include the provision of new or modified parameters to be passed to the failed O/S image's boot loader. For example, in the case of Windows O/S, it may need to be started in "safe mode", or, possibly, to change the kernel that is to be used, modify the number of CPUs the O/S should recognize, turn on boot logging, or, specify a different memory size. This system is most practically applied when there exists a network connection between the computer running the diagnostic software package and the proxy system. This allows data to be easily transported between the two systems without human intervention.

[0012] Due to the communication network connection, the proxy system may be remotely located from the computer running the diagnostic software package and hence, the owner of the proxy system may provide a computer diagnostic repair service.

[0013] Figure 1 depicts generally the diagnostic repair system 10 according to the invention. As shown in Figure 1, there is depicted a first system 20 to be diagnosed and repaired that includes a computing node, workstation, PC or any processing device implementing an Operating System (O/S) such as Windows, UNIX, etc. As depicted in Figure 1, the O/S may reside in a memory, e.g., a hard disk, and includes a Primary OS partition 25 that includes a customer's O/S (e.g., Windows 2000 or like operating system). A second partition 30 in the memory includes a diagnostic O/S (e.g., a Linux-based, or a DOS-based operating system, etc). As described herein with respect to Figure 1, it is assumed that, for purposes of discussion, the primary customer operating system executing on the first partition 25 is a Windows-based O/S while the diagnostic application residing on the second partition 30 is Linux-based. When the

primary operating system fails (e.g., a “blue” screen is displayed, for example) the diagnostic O/S 30 is automatically booted and is executed as shown as the running diagnostic O/S 35 in Figure 1. Preferably, according to the invention, the diagnostic O/S 35 accesses the primary O/S partition 25 to retrieve diagnosis related files 32, for example registry, system files, system settings, error files, error logs, IP addresses, and other system configuration files, etc. It is understood that some of these collected files may comprise information proprietary to the failed O/S and unable to be read by the diagnostic O/S 35. Thus according to the invention, the system further includes a communication link 50 enabling communications between the diagnosed system 20 to a proxy diagnostic system, e.g., a server device 60. Thus, as shown in Figure 1, the diagnosed system 20 is running a diagnostic O/S 35 (e.g., Linux) and the proxy diagnostic server 60 is running the same as the primary O/S 25 (e.g., Windows) that was running on the diagnosed system 60. The system being diagnosed 20 sends the diagnosis related files 32 to the proxy diagnostic server 60 over communication link 50 (e.g., TCP/IP, Ethernet, or like “low-level” network connections, etc.) because the diagnostic O/S 35 (e.g., Linux) does not understand some of the file formats of the primary O/S (e.g. Windows) files 32. A primary O/S 65 executing on the proxy diagnostic server 60 having a hard disk drive partition 75 is of the same O/S format as the primary O/S on the system 20 being diagnosed, and receives the diagnosis related files 32 from the diagnosed system 20. The proxy diagnostic server 60 examines the diagnosis related information sent from the diagnosed O/S system 20 and, uses tools only available under primary O/S (e.g. Windows regedt32, Event Viewer, memory dump analyzer), and not available under the diagnostic O/S (e.g. Linux) 35. The proxy diagnostic server 60 determines the problem and sends the repair information, and if possible, a fix (a service patch) 37, for example, over communication link 50 back to the system 20 being diagnosed. The system 20 being diagnosed receives the service patch 37. The executing diagnostic O/S 35 writes the service patch to the primary O/S partition 25, takes the necessary repair actions, and shuts down itself so that primary O/S may again start. It should be understood that it is also possible to perform the repair by having the diagnostic O/S e-mail the configuration data and then await an e-mail reply from the proxy diagnostic O/S.

**[0014]** A practical example of the diagnostic repair system 10 according to the invention illustrated in Figure 1 is now described in the following example: Assume that pre-Boot Linux is used as the hosting environment 35 for the diagnostic tool which is reasonable since it has a small footprint, network connectivity, and is free. If a computer system fails that is running Windows O/S 25, the Linux-based diagnostic software is able to verify that it is not a hardware failure. However, it cannot perform a detailed examination of the Window's system configuration (registry, event log, security descriptors, upcoming Windows system components, etc.) as the data is stored in undocumented proprietary formats that need to be manipulated with interfaces that are only available under a running Windows system. However, as the files that Windows generates and stores in the course of normal operation may be read by Linux, the Linux-based diagnostic tool may collect the configuration information and send it off over the network to a proxy Windows system 65 for processing. A repair application is easily written to run on the proxy server 60 running the same O/S as the failed image. One necessary requirement is that the secondary system be enabled to examine the collected configuration data of the failed O/S image using the available Windows interfaces to determine problem causes and corrections. The repair application executing on the secondary server 60 will then send a repair file back to the Linux Diagnostic software that includes modified configuration data and replacement files with additional instructions to copy or overwrite files that may fix the failed O/S and/or update the disk accordingly. The Linux-based O/S may then shut down as the failed primary O/S (e.g., Windows) is re-booted.

**[0015]** In a further embodiment, it is understood that an organization may provide a computing system diagnostic service for repairing failed operating systems. For instance, a remote service organization may comprise a number of proxy server devices each running one of many types of operating systems. Thus, the entity with the failed primary operating system will boot the hidden diagnostic repair application (e.g., Linux) to obtain the relevant configuration files of the failed O/S image and send these files to the service organization which includes a proxy device executing the same O/S as the entity. In the manner described, a repair file may quickly and easily be generated by the service party to fix the failed entity's computer O/S which file may be communicated back to the failed system over a network connection.

**[0016]** It should be further understood that there is a cost advantage in implementing the system 10 of the present invention, particularly, as the system avoids the need for a company that provides the Linux diagnostic software to purchase an additional Window's license for each system that it is supporting. For example, in a large organization, only a small number of additional Windows licenses will be needed to be purchased for the proxy devices compared to the total number that would need to be purchased, for example, had Windows (a primary O/s) been chosen as the pre-Boot environment for the diagnostic software package.

**[0017]** While there has been shown and described what is considered to be preferred embodiments of the invention, it will, of course, be understood that various modifications and changes in form or detail could readily be made without departing from the spirit of the invention. It is therefore intended that the invention be not limited to the exact forms described and illustrated, but should be constructed to cover all modifications that may fall within the scope of the appended claims.